Estimation of Travel Times for Minor Roads in Urban Areas Using Sparse Travel Time Data

Luong H. Vu, Benjamin N. Passow, Daniel Paluszczyszyn, Lipika Deka, and Eric Goodyer

***Abstract*—Travel time is a basic measure based on which intelligent transportation systems such as traveller information systems, traffic management systems, public transportation systems are developed. Although many methodologies have been proposed, they have not yet adequately solved many challenges associated with travel time, in particular, travel time estimation for all links in a large and dynamic urban traffic network is still an open problem that needs addressing. Typically focus is placed on major roads such as motorways and main city arteries but there is an increasing need to know accurate travel times for minor urban roads. Such information is crucial for tackling air quality problems, accommodate the growing number of cars and provide accurate information for routing. This study aims to address the aforementioned challenges by introducing a methodology, namely Similar Model Searching (SMS), to estimate travel times by using historical sparse travel time data. The SMS learns the temporal and spatial relationship between the travel time of adjacent links and utilise labelled data of similar models in order to improve its overall performance. The effectiveness of the proposed method is evaluated on a section of Leicestershire traffic network in the UK. The obtained results show that SMS efficiently estimates travel time of target links using models of adjacent traffic links.**

***Index Terms*—travel time, sparse data, machine learning, traffic model, temporal, spatial.**

I. Introduction

TRAFFIC congestion can be defined as the traffic demand exceeding the roadway capacity. Congestion is becoming increasingly problematic issue for major cities across the globe. According to [1] in the United Kingdom the estimated cost of congestion in 2017 was more than £37.7 billion; with London ranked the 7th most congested city in the world.

While a number of works were undertaken to increase transport networks' capacity, in urban areas, transportation infrastructure development is constrained by land and financial resources [2]. Another approach to deal with congestion is by improving the current traffic management strategies [3]. However, to effectively respond to daily traffic challenges operators need current travel times data or accurate models of travel time.

Furthermore information from the travel time model can be useful to: commuters to make efficient travel decisions such as for route choice, mode of transport and time of travel; traffic policy sector in forecasting travel demand and evaluation of the impact of policy instruments, e.g. congestion charges [4]; vehicle routing problems [5].

Travel time can be measured and collected typically by using stationary or moving observers. Stationary observers include loop detectors and video surveillance, which provide flow and speed estimation at regular and frequent intervals. Moving observers, including floating cars (data collected as vehicles do normal trip), probe cars (used explicitly for collecting data), vehicle fleets with GPS devices or smartphones, provide information which can be used to extract travel time data in road segments where the moving observers go through [6]. Travel time data source directly influences the properties of travel time data. The stationary observers can collect travel time data at regular and frequent intervals. However, the share of segments in the network equipped with these observers is typically low and not representative of the urban network as a whole, which leaves the traffic conditions in most of the network unknown [7]. Similarly, the moving observers can collect travel time data at irregular, less frequent intervals and in limited duration of time, which means that, at some times of a day there might be no travel time data available for a particular road segment. Also, the moving observers enable collection of travel time information across the entire urban road network [6], [8].

Travel time data on motorways regularly show relatively low variability (the variabilities are less than 3.5 seconds/km [9]), especially in congested conditions. The enforced speed limit reduces the speed difference between vehicles, which results in a lower travel time variability. [9] indicated that the travel time variability mainly depends on geometrical characteristics of motorways; e.g. the number of ramps weaving sections per unit road length (ramps refer to interchanges which permit traffic on a motorway to pass through the junction without interruption from any other traffic stream, the number of lanes etc.). In contrast, the urban travel times can be subject to very high variability because of traffic light signal cycles and queue delays. Pedestrians and cyclists and on-street parking also affect travel time, [6], [10]. This poses a challenge to design models or algorithms that can estimate accurately near real-time travel time in urban areas.

In [11] the Neighbouring Link Inference Method (NLIM) was introduced to deal with the highly sparse data collected from moving observers in a large urban traffic network. The NLIM learns the relationship between travel time of a road segment (link) and traffic parameters (travel time, vehicle class, time of day, day of week) of its nearby links using feed forward back propagation neural network. Subsequently, the NLIM model is used to estimate near real-time travel time for

Luong H. Vu, Benjamin N. Passow, Daniel Paluszczyszyn, Lipika Deka and Eric Goodyer are with the De Montfort University's Interdisciplinary Research Group in Intelligent Transport Systems (DIGITS), De Montfort University, Leicester, United Kingdom e-mail: vuhuyluong@gmail.com.

Daniel Paluszczyszyn was with Wroclaw University of Economics, Komandorska 118/120, 53-345 Wroclaw, Poland.

links, which do not have recently observed travel time data. An outlier detection based on the Gaussian mixture model was proposed to remove anomalies from travel time data. Results in [11] demonstrated that the NLIM method outperforms the statistic-based and linear least square estimation methods. However, it was observed that the NLIM performs better on major links than minor links; it produces higher Mean Absolute Percentage Error (MAPE) for the minor links than for the major links.

As a substantial extension to methods described in [11], this paper aims to improve the performance of the NLIM, especially for the minor links, as the vast majority (i.e. 70%) of links in the UK fall within the minor link category [12]. Compared to work in [11] this paper contains the following new contributions: i) the new Similar Model Searching (SMS) algorithm; ii) utilisation of travel time data of similar models to improve relationship between links in a target model; iii) a much larger case study area and additional performance indicators. Our results demonstrate that the proposed SMS method can work more effectively with highly sparse data and improve the performance of the original method especially for the minor links.

The reminder of this paper is organized as follows. Section II reviews the related work. The details of the proposed algorithm are given in Section III, followed by Section IV that evaluates the performance of the SMS, and conclusions are provided in Section V.

II. Related Work

Accurate travel time estimation is crucial for efficient urban road network operation but it is a challenging subject in the intelligent transportation system as different delays from traffic signal controls, congestion effects, stochastic incidents, etc., are introducing many uncertainties into travel time data [13].

Travel time estimation is defined as the method which approximates the travel time of vehicles on a given link during a given period. The existing travel time estimation methods can be classified as direct or indirect methodologies [14]. In the direct method, travel time is estimated based on data samples that are obtained from moving observers e.g. in-car sensor equipment [15], [16], GNSS-based floating car [17], [18], automated vehicle identification system [19], telecommunication activities [20], [21].

The advantage of the direct method is that it requires limited expenses of infrastructure and it is capable of producing travel time data in small roads where loop detectors may not be deployed. The drawback of the direct method is that for example a car cannot collect data in different locations simultaneously. Also at different times, the particular road may exhibit different dynamics which may not be captured by a probe car. Hence, uncovering a methodology for travel time estimation from incomplete datasets receives a great interest from researchers in the field of the Intelligent Transportation Systems (ITS).

The indirect method uses data obtained by stationary observers, e.g. inductive loop detectors [22], to analyse the correlation between travel time and traffic flow. The inductive loop detectors are usually deployed at junctions and segments of major roads. The indirect method can provide travel time data at a regular sampling rate.

The majority of travel estimatiom methods use statistical and mathematical techniques [4], [5], [13], [17]. Mathematical and statistical methodologies usually perform less accurate in urban traffic network where the traffic condition can be complex. There are also approaches that utilise artificial neural networks [14], support vector machines [23], linear regression [23] and non-linear least square [24]. They can learn relationships and create models using unstructured dataset. The approaches are often useful in many transportation applications because they are free of model assumptions and the uncertainty of traffic can be involved in the traffic model. These models do not include temporal and spatial dependencies and hence not always accurate.

Researchers have recently explored the use of deep learning techniques in the field of ITS [25] and have obtained very promising results. However, data in the context of the problem addressed within this paper are highly irregular and sparse and deep learning techniques are not always the best and obvious choice.

Several studies, including [4], [8], [22], explored temporal and spatial dependencies in traffic. The integration of temporal and spatial relationships of traffic information into traffic models could enhance their estimation capabilities [4].

[26], [27] proposed prediction methods that using similarities in traffic pattern for links in a traffic network. In [27], express-way travel time is predicted based on matching real-time traffic patterns to historical pattern. In [26] traffic flow is predicted based on similar profiles from the historical data. Both methods explore temporal relations between links in a traffic network.

An approach of applying temporal and spatial dependencies in travel time estimation was presented in [22]. The temporal-spatial queueing uses headway travel time series, which are collected from upstream and downstream of a middle link, and a recent vehicle speed to estimate the middle link's travel time data. The model utilises the relationship between upstream travel time and downstream travel time to enhance the accuracy of travel time estimations. The proposed method can model fast travel time variations. In [8] traffic data of nearby links is used to forecast travel time of a selected road segment. The method was termed as geospatial inference. Both studies used travel time data series which naturally have the temporal relationship.

[4] proposed a purely data-driven approach, namely, a tensor-based citywide spatial-temporal travel time modelling. The proposed method utilises the spatial-temporal approach in modelling the travel time of all traffic links under different traffic conditions and time slots. The methodology is complicated because of characteristics of tensor-based techniques as well as the correlation between travel times and the influential factors on the complexity of urban traffic networks. The concept proposed [4] is that similar traffic condition in the traffic link should produce a similar travel time for a specific driver. The centroid of the cluster represents the travel time of the corresponding traffic condition and the corresponding driver.

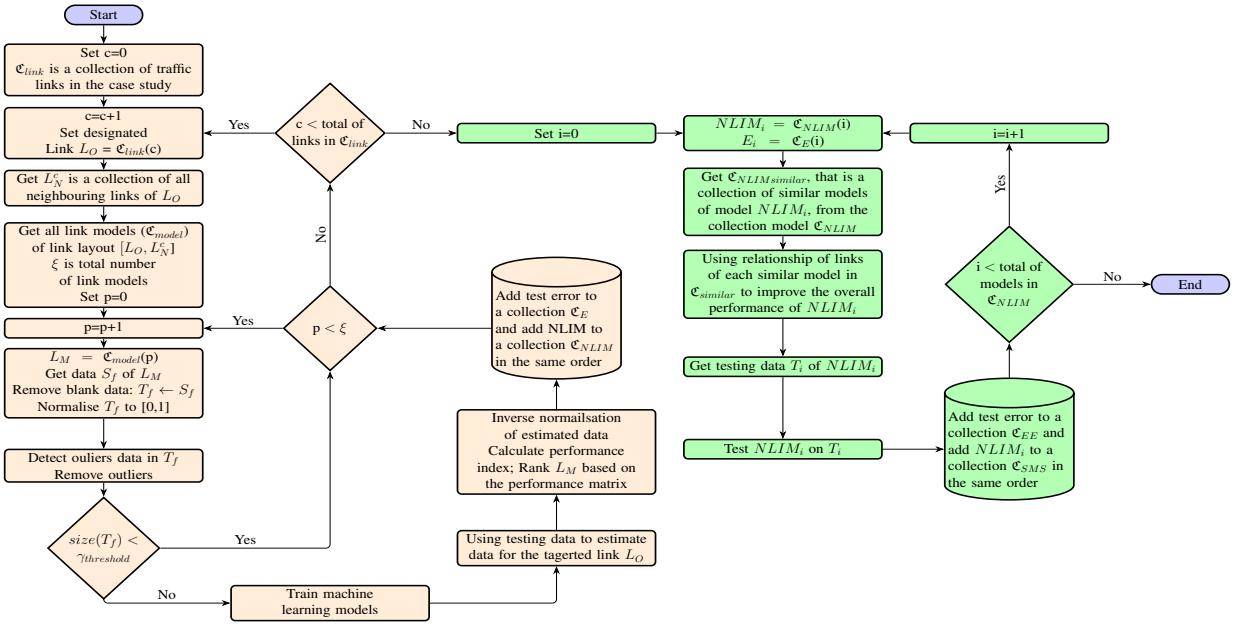


Fig. 1. Diagram of Neighbouring Link Inference Method with Similar Models Searching. The NLIM is shown as the orange blocks while the green blocks represent the SMS methodology. $\gamma_{threshold}$ is the minimum number of labelled data.

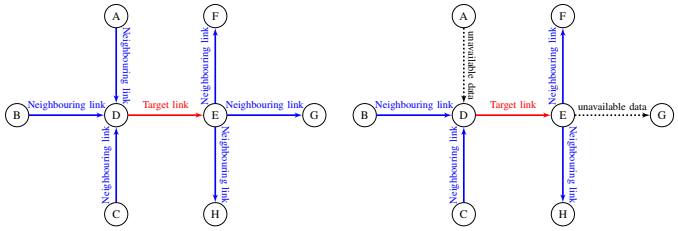
Based on the current traffic condition, a corresponding cluster of historical travel times is selected. The missing travel time is replaced with the centroid of the cluster. The advantage of the method is that the travel time can be easily modelled as a 3-order tensor despite the complexity of urban traffic network, and the technique can work with high data sparsity but it has high sampling rates. The method does not express the relationships between links in travel trajectories and those on traffic links of two different travel time trajectories. The travel time in the clusters is selected based on the time slot, corresponding driver and corresponding traffic link; thus, travel times seem to have temporal relationship only.

In [11], the NLIM was introduced to deal with the datasets with high sparsity and irregularity, which have entries only for major links or entries collected at highly irregular intervals. Having embedded knowledge about the temporal and spatial dependencies between travel times of a target link and its adjacent links the model can overcome sparsity in input data and provide accurate estimations. The subsequent sections describe the research carried out to improve the performance of the NLIM, especially in minor links. The travel time estimation for minor links is recently receiving more interest due to upcoming autonomous vehicles and more integrated ITS. For clarification, Fig. 1 depicts steps of the NLIM method. More details on the NLIM can be found in [11], [28].

III. Similar Model Searching Method

A. Definitions

1) Traffic Link Classification: Different road categories produce different traffic travel times. In this research classification proposed by [12] is adopted. Furthermore, the major link refers to a combination of the motorway, trunk, primary and A link. The minor link refers to the remaining road categories.



(a) Link model, size=6 (b) Link model, size=4

Fig. 2. Examples of different traffic link models from a traffic link layout. The solid arrows represent links with data available where the dashed arrows represent links which do not contain data.

2) Traffic Link Layout and Traffic Link Model: Transportation systems which include structure and flows are commonly represented using networks as an analogy. It is a sub-category of the spatial network since transport networks' design and evolution are physically constrained. A traffic links layout is a simplified representation of a small part within the traffic network. It depicts junctions and roads. However, roads are presented as unidirectional connections between the junctions to indicate a traffic flow direction. A node is a traffic link network term which indicates intersections in the transportation network, [29] and the structure of links within a locations' system. A traffic link is a single direct route between two nodes in a network, [29], [30].

For clarification, a traffic link layout used in this paper is shown in Fig. 2a. It comprises seven directional connections based on the assumption that traffic-related information in the rear connections (AD, BD, CD) and the front connections (EF, EG, EH) affect those in the middle connection (DE). Note, link reference say DE indicates the direction of traffic is from node D to node E.

Traffic links layout consists of a targeted link and adjacent

links. The target link is a link where traffic-related information needs to be determined. The neighbouring links are links that might contain information that can be used for the traffic parameters estimation.

In the model shown in Fig. 2a, DE is the target link ($L_O = \{DE\}$) and AD, BD, CD, EF, EG, EH are neighbouring links ($L_N^{DE} = \{AD, BD, CD, EF, EG, EH\}$). Specifically, AD, BD, CD are the rear neighbouring links ($L_{NR}^{DE} = \{AD, BD, CD\}$) and EF, EG, EH are the front neighbouring links($L_{NF}^{DE} = \{EF, EG, EH\}$) of link DE. In Fig. 2a, there are 6 neighbouring links in total. $[L_O, L_N^{DE}]$ denotes the link layout.

A traffic link is a neighbouring link if it shares a node with the target traffic link. The beginning node of a front link is the end node of its target traffic link, and the end node of a rear link is the beginning node of a target traffic link.

The traffic link model consists of a target link and at least one of the adjacent links (i.e. the links with available data on required traffic parameters as explained below). A full traffic link model includes the target link and all of the neighbouring links. If a traffic link layout has N neighbouring links, then the total number of traffic link models ξ can be calculated as follows:

$$\xi = \sum_{k=1}^{N} \frac{N!}{k!(N-k)!} \tag{1}$$

L_N denotes a set of neighbouring links in a traffic link layout, L_M denotes a set of neighbouring links in a specific traffic link model ($L_M \in L_N$). Fig. 2 depicts two examples of traffic link models with a different number of neighbouring links. The solid arrows represent the links that are included in the traffic link model. The traffic link $L_O = DE$ is the target link. Fig. 2a displays a full traffic link model of the link layout that includes all neighbouring links($L_M^{DE} = L_N^{DE} = \{AD, BD, CD, EF, EG, EF\}$). Fig. 2b shows a traffic links model when only a subset of data is available for constructing a model. It is important to highlight that created models may have different number of links as data for traffic links might not always be available at the required time. Only a small portion of real-world traffic networks is monitored; mainly major roads. The proposed approach addresses this problem by creating a set of traffic link models based on the links that contain data. The accuracy of traffic parameters estimation depends on the availability of data and degree of the relationship between links that can be detected.

Travel time data are coded in a matrix form where each entry represents whether travel time of a vehicle type is present on a specific day of a week and at particular time interval in the day. The vehicle class v ranges from 1 to 9. The day of a week d has a value from 0 to 6 that represents Monday to Sunday, respectively, and the time interval is denoted t.

Define S as a data matrix of a traffic link layout which includes data of a target link and data of all its neighbouring links, S^{in} is the data matrix of the neighbouring links and S^{out} is the data matrix of the target link.

The structure of a data matrix S_f for the full traffic model ($L_O = \{DE\}$ and $L_M^{DE} = \{AD, BD, CD, EF, EG, EH\}$), shown in Fig. 2a, is presented below:

$$\begin{cases} S_f = \Big[[d], [t], [v], [t_{AD}], [t_{BD}], [t_{CD}], [t_{DE}], [t_{EF}], [t_{EG}], [t_{EH}]\Big] \\ S_f^{in} = \Big[[d], [t], [v], [t_{AD}], [t_{BD}], [t_{CD}], [t_{EF}], [t_{EG}], [t_{EH}]\Big] \\ S_f^{out} = \Big[t_{DE}\Big] \end{cases} \tag{2}$$

where S_f^{in} and S_f^{out} are the model's input features and output feature subsequently forwarded to machine learning. $t_{AD}, t_{BD}, ..., t_{DE}$ are travel times of v for the corresponding traffic link on a specific day and time (d, t). If the travel time data does not exist at the specific time the value of the corresponding entry is set to blank.

Some machine learning techniques use labelled data to generalise the relationship between input and output data. If the data has empty entries, many machine learning techniques cannot utilise these instances for modelling. Therefore, in this work, all blank/empty entries need to be removed before the dataset can be used for training and testing. The matrix S is transformed into the matrix T (T_{full}, T_{full}^{in}, T_{full}^{out}).

3) Data sparsity: The sparsity of the data matrix corresponding to a traffic link model is defined as a ratio between the number of empty entries to the total number of elements in the matrix. In this paper, the sparsity of a dataset is a measurement indicator. The lower the sparsity of the dataset, the higher amount of available travel time data from moving observers in the traffic link model.

B. Similar Model Searching

In this section, a novel SMS methodology is introduced to deal with high data sparsity and irregularities in traffic network data. The SMS learns the temporal and spatial relationship between the travel time of adjacent links and uses this relation to estimate travel time of the targeted link. For this purpose, several machine learning techniques including support vector machine regression, neural networks and multi-linear regression are employed.

The SMS method takes as input models derived by the NLIM. NLIM uses a feed-forward back propagation neural network to derive models describing the relationship between the target links travel time and the traffic parameters of its neighbouring link travels time, vehicle model, time of day etc. The main idea of SMS is to discover a list of traffic link models which has similarity with a target traffic link model. Within SMS, two links are considered similar if the models of the relationship between the links and their respective neighbours are similar.

The target model $\{L_O, L_N\}$ is similar to a model $\{\bar{L}_O, \bar{L}_N\}$ if they satisfy two conditions:

1) The size of L_N is equal the size of $\bar{L}_N$, where size refers to the number of neighbouring links in a traffic link model.
2) The relationship between L_O and links in L_N is similar to the relationship between $\bar{L}_O$ and links in $\bar{L}_N$.

Condition 1 is trivial to confirm while Condition 2 needs to use the model of $\{L_O, L_N\}$, generated using the NLIM technique, to examine the model $\{\bar{L}_O, \bar{L}_N\}$. For the models to be similar the error of $\{\bar{L}_O, \bar{L}_N\}$ must be less than or equal

to error of $\{L_O, L_N\}$. For both models the error is calculated based on test dataset of $\{L_O, L_N\}$ because lesser error using same data by $\{\bar{L}_O, \bar{L}_N\}$ means that $\{\bar{L}_O, \bar{L}_N\}$ potentially has similar relationship between target link and the neighbouring links as $\{L_O, L_N\}$.

If the similarity conditions are satisfied the training dataset of similar models can be also used as training dataset for the target model. It is to be noted that the link length in the traffic link model, the shape of the traffic link layout and the shape of the traffic link model are not directly considered as conditions in the SMS because they are already included in the link relationships. The steps of SMS are presented in Algorithm 1. The SMS is not a standalone method and requires a collection of models obtained with use of the NLIM. The input to the SMS algorithm is a collection of NLIM models $\mathfrak{C}_{NLIM}$ and the corresponding errors $\mathfrak{C}_E$. Fig. 1 illustrates dependencies and distinctions between the SMS and NLIM. More details about the particular section of the algorithm can be found in [11], [28].

As mentioned in the previous section, the number of data samples in each model is not identical. The motorway, trunk and primary links may have a large amount of travel time data that can be used for training and testing while A links, B links, and minor links may have a lower number of data samples. Consequently, the performance of the models might be affected. The proposed SMS methodology can be applied to address the insufficient number of data samples for minor roads.

Algorithm 1 Similar Model Searching

function SMS($\mathfrak{C}_{NLIM}$, $\mathfrak{C}_E$)
  $\tau \leftarrow$ number of models in $\mathfrak{C}_{NLIM}$
  for i=1 **to** τ **do**
    do
      $NLIM_i \leftarrow \mathfrak{C}_{NLIM}(i)$
      $I_i \leftarrow$ number of $NLIM_i$'s inputs
      $T_i \leftarrow$ training data of $NLIM_i$
      $T_i \leftarrow normarlise(T_i)$
      $T_i \leftarrow$ DR-M-GMM(T_i, ϵ, k)
      $T_i' \leftarrow$ testing data of $NLIM_i$
      $Error_i \leftarrow \mathfrak{C}_E(i)$
      for j=0 **to** τ **do**
        $NLIM_j \leftarrow \mathfrak{C}_{NLIM}(j)$
        $Errror_j \leftarrow \mathfrak{C}_E(j)$
        $I_j \leftarrow$ number of $NLIM_j$'s inputs
        if i $\neq$ j **and** $I_i = I_j$ **then**
          Insert $NLIM_j$ into $\mathfrak{C}_{PS}$
          Insert $Error_j$ into $\mathfrak{C}_{PE}$
        end if
      end for
      $\bar{T}_i \leftarrow T_i$
      Sort $\mathfrak{C}_{PS}$ in descending order based on $\mathfrak{C}_{PE}$
      $Sk_{PS} = 0$
      for each $NLIM_{PS}$ **in** $\mathfrak{C}_{PS}$ **do**
        $Error_{PS} \leftarrow$ Error of $NLIM_{PS}$ on T_i'
        if $Error_{PSimilar} \leq Error_i$ **then**
          $T_{PS} \leftarrow$ training data of $NLIM_{PS}$
          $T_{PS} \leftarrow normarlise(T_{PS})$
          $T_{PS} \leftarrow$ DR-M-GMM(D_{PS}, ϵ, k)
          $\bar{T}_i \leftarrow T_i + T_{PS}$
          $(\bar{T}_i^{in}, \bar{T}_i^{out}) \leftarrow \bar{T}_i$
          $n \leftarrow 1000$ ▷ the number of labelled data for hyper-parameter searching
          $\Theta \leftarrow \emptyset$
          $GRIDSEARCH(\bar{T}_i^{in}, \bar{T}_i^{out}, n, \Theta)$
          $NLIM_i' \leftarrow$ LEARNING($\bar{T}_i^{in}, \bar{T}_i^{out}, \theta_{best}$)
          $Error_i'$ of $NLIM_i'$ on T_i'
          if $Error_i' \leq Error_i$ **then**
            Insert $NLIM_i'$ into $\mathfrak{C}_{SMS}$
            Insert $Error_i'$ into $\mathfrak{C}_{EE}$
            $T_i \leftarrow \bar{T}_i$
            $Sk_{PS} \leftarrow 0$
          else
            $\bar{T}_i \leftarrow T_i$
            $Sk_{PS} \leftarrow Sk_{PS} + 1$
          end if
        end if
      end for
    while $Sk_{PS} \geq 3$
  end for
end function

IV. Case Study: Leicestershire Traffic Network

A. Experimental Data

The evaluation was carried out on a Floating Car Data (FCD) dataset courtesy of Teletrac (formerly Trafficmaster). Teletrac provide a cloud-based GNSS tracking software for fleet tracking [31]. The travel time data was collected from September 2009 to February 2012 in Leicestershire, UK. The dataset (approx. 60Gb) is in a CSV file format and consists of data for an individual link on a monthly basis. The dataset contains 240000 traffic links but as this work is focused on the urban areas only, the dataset used in the subsequent evaluations comprises travel times for 22053 traffic links. The selected area shown in Fig. 3 includes Leicester (major city) and its surroundings. Data sparsity of the links is present by colours. The range of colour from black to green indicates data sparsity from 100% to 0%. The case study statistics are listed in Table I. The FCD dataset contains reconstructed link travel times at 15 minute intervals. A day starting from 00h00 to 23h59, was divided into 96 time slots. The average travel time for links in the traffic network is approximately 2.46 (minutes/miles). Total links' length is roughly 14,000 kilometres (8,700 miles). There are 9 vehicle classes which are based on the payload and the size of the vehicle.

Fig. 4 and Fig. 5 gave further insight into the complexity, irregularity and sparsity of the dataset. Fig. 4 indicates that 69.42% of 240000 links in the full dataset have sparsity $\leq$ 99%. Fig. 5 shows the travel time samples distribution across a day, which likely resembles the daily activity of probe car drivers.

TABLE I
The Number of Links in the Dataset

Link type	Number of links
Motorway	67
Trunk	22
Primary	911
A	1457
B	843
Minor roads (data sparsity $\leq$ 99%)	5226
Minor roads (data sparsity $>$ 99%)	8526
	Total: 22053

The case study traffic network shown in Fig. 3 and Table II indicates the sparsity distribution amongst the links; the dataset is more sparse on the urban traffic links than on the motorway links. The lower quartile, the median and the upper quartile of data sparse rate on motorway links are 54.9%, 19.5% and

TABLE II
EXPERIMENT DATA SPARSITY (%) PER LINK TYPE.

	Motorway	Trunk	Primary	A	B	Minor Road
Lower whisker	0.0	20.0	36.9	0.0	39.9	68.6
Lower quartile 25%	10.5	40.2	70.7	76.8	83.7	92.9
Median	19.5	74.6	77.6	81.3	87.1	95.5
Upper quartile 75%	54.9	81.5	85.0	85.9	90.5	97.7
Upper whisker	93.1	98.7	100	100	100	100

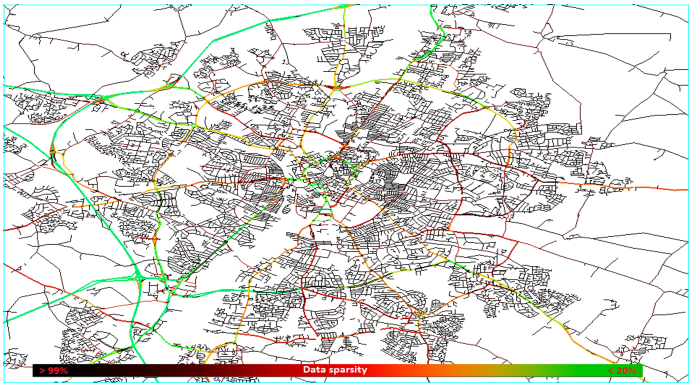


Fig. 3. Illustrating the area and data sparsity in the dataset for the case study traffic network.

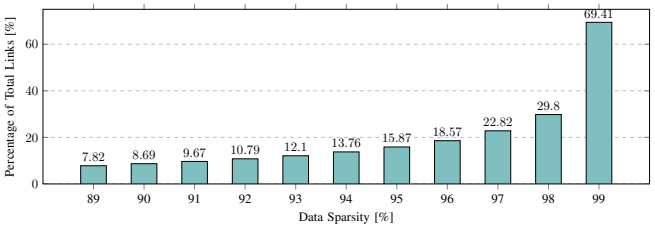


Fig. 4. Data sparsity distribution in links with data sparsity $\leq$ 99% in the Leicestershire traffic network calculated for the full dataset.

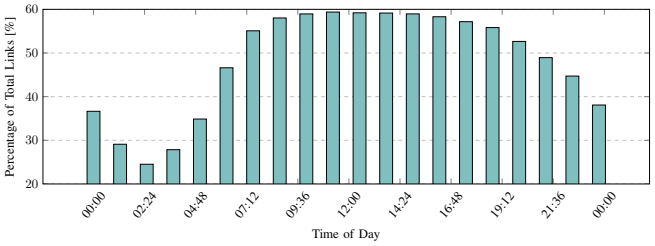


Fig. 5. Data sparsity distribution across a day for links with data sparsity $\leq$ 99% in the Leicestershire traffic network.

10.5% respectively. Their values are significantly greater on the urban links.

To account for the sparsity a Data Sparsity Threshold (DST) is introduced to qualify only the links with sparsity less than or equal to the DST in the experiment. Therefore, the number of links, link layouts, possible models in the experiment will be dependent on the DST value. The links involved in an experiment at specific DST value are named DST-Links. The subsequent sections will provide suggestions how to determine the DST value.

B. Experiment Settings

The SMS and NLIM models were trained on the case study dataset employing multivariate linear regression, feed forward evolution learning neural network and feed forward resilient back propagation neural network (NLIM-MLR, NLIM-EL and NLIM-RPROP, respectively). Input features for training and testing models are sparse historical (2009-2012) travel time data of neighbouring links, corresponding time of day (time slot), vehicle class and day of the week. The output feature is the corresponding travel time on the target link. The models were trained and validated to make sure relationships between temporal and spatial of travel times in links is captured. The accuracy of models is evaluated by Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and MAPE performance metrics on unseen data. More details on training procedure can be found in [28].

The SMS models are compared against models obtained by NLIM-MLR, NLIM-EL, NLIM-RPROP, Historical Travel Average (HA) and Moving Average (MA) methods. HA and MA are classical methods estimating the current travel time by using historical travel time data [4]. HA uses the corresponding average of the historical travel time of a time slot on a target link to estimate the current time slot travel time on the link. Meanwhile, MA uses moving average of three-time slots right before the current time slot to estimate the current time slot travel time.

The case study dataset has 13527 links which have data sparsity $\leq$ 99%. These links were involved in the experiment. They represent approximately 61.34% of total traffic links in the experiment area. The SMS requires a collection of NLIM models and the corresponding errors. In this case 338177 link models were created by the NLIM from 13527 links. Each target link has a combination of the NLIM models. The diversity of models' size and relationships between traffic links gives a possibility of having many potential similar models in the collection.

C. Results

The SMS searches for similar models among the 338177 NLIM models (see Algorithm 1). Once similar models are found, the SMS does a further step to check if training data of the potential similar models can be adapted to enhance the performance of the selected NLIM model. By using SMS, the NLIM model does not only utilise data of its link model but also of other similar link models in the traffic network. This effectively strengthen the temporal and spatial relationship between travel times in links of the target link model.

Fig. 6 and Fig. 7 demonstrate the effectiveness of the SMS in increasing the number of training data samples. The comparison is with respect to the original training dataset. After the SMS was applied to the 338177 NLIM travel time models the mean number of increased training samples was 22439, 22613 and 26618 for minor roads, B and A links respectively. While those on primary, trunk and motorway links were 18952, 16865 and 20408. It can be seen that SMS works more effectively on the minor links than on the major links.

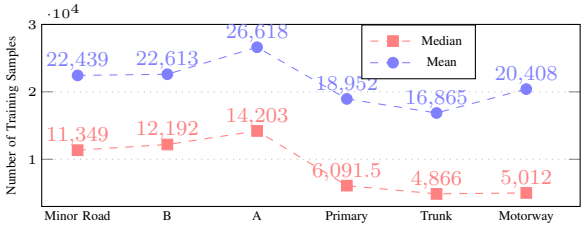


Fig. 6. The mean and median of the increased number of training samples per link type.

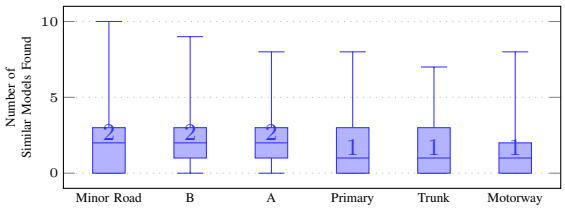


Fig. 7. The number of similar NLIM models found by SMS per link type.

TABLE III
THE PERFORMANCE METRICS OF SMS, NLIM-MLR, NLIM-EL, NLIM-RPROP, MA AND HA MODELS ON UNSEEN DATA: (1) LOWER-WHISKER, (2) LOWER-QUARTILE,(3) MEDIAN, (4) UPPER-QUARTILE, (5) UPPER-WHISKER

Model	(1)	(2)	(3)	(4)	(5)
RMSE [seconds]					
MA	0.23	1.64	3.39	9.12	473.51
HA	0.12	2.43	5.78	12.76	625.03
NLIM-MLR	0.12	1.62	4.18	12.63	453.45
NLIM-EL	0.04	1.41	3.13	7.85	548.28
NLIM-RPROP	0.04	1.48	3.25	8.10	548.15
SMS	0.02	1.03	2.37	6.06	275.38
MAE [seconds]					
MA	0.13	0.91	1.71	3.52	310.19
HA	0.08	1.22	2.65	5.60	454.30
NLIM-MLR	0.15	0.84	1.96	5.10	830.26
NLIM-EL	0.02	0.76	1.63	3.54	380.59
NLIM-RPROP	0.02	0.80	1.72	3.74	424.95
SMS	0.01	0.53	1.17	2.63	130.96
MAPE [%]					
MA	12.67	25.02	30.01	38.86	403.31
HA	12.41	22.89	30.83	45.73	401.26
NLIM-MLR	8.03	18.24	24.69	40.31	7894.34
NLIM-EL	3.07	12.72	17.15	25.78	910.30
NLIM-RPROP	1.26	13.42	18.08	27.14	3177.59
SMS	0.804	9.52	13.59	19.56	428.90

Table III presents outcomes of the SMS, NLIM, MA and HA methods using the five-number summary in terms of the adopted performance metrics. The performances of SMS models are significantly better to those achieved by NLIM-MLR, NLIM-EL, NLIM-RPROP, HA and MA. It is worth to highlight that the performances of NLIM-based methods are also more accurate than those of HA and MA. It has also been seen that 75% of the best model in terms of MAPE for the methods NLIM-MLR, NLIM-RPROP and NLIM-EL have much higher RMSE as well as MAE values compared to the SMS method.

As demonstrated in Table II the sparsity of links used in the experiment varied (0%-100%) in each link type. The sensitivity of the evaluated methods, SMS, NLIM-EL, NLIM-RPROP and NLIM-MLR, to the DST value was also investigated. The results in Fig. 8 clearly show that the DST has an impact on the number links involved in the experiment.

It can be seen in Fig. 8 when DST was set to a shallow value (i.e. DST=0%-50%), the number of DST-Links is less than 5%. The number of DST-Links is greatly increased from over 10% to over 60% when DST value increases from 80% to 99%. However, the number of the best traffic link models that have MAPE $\leq$20% decreases from approximately 70% to under 20%, including SMS. But the number of SMS models which have MAPE $\leq$20% is always higher, i.e. between 5% and 10% than those of NLIM-EL, NLIM-RPROP and NLIM-MLR.

The same trends can be observed in Fig. 9. The number of the best traffic link models that have RMSE $\leq$3 seconds is also a notably decreased from approximate 60% to under 35% when DST value rises from approximate 70% to 99%. Still, the number of SMS models which have RMSE $\leq$3 seconds is noticeable higher than other methods.

The performance of SMS was evaluated regarding a very high data sparsity (DST=99%) to show the ability of SMS in modelling the links. For the threshold of 99%, the number of DST-Links was 13527, and the number of traffic link models was 338177. According to the statistics in Table III, more than 75% of the best SMS models have MAPE less than or equal to 19.56%.

It can be determined from Fig. 8 that, the SMS method has the best performance at DST =70% in terms of improving the percentage of target links that their travel time can be accurately estimated by SMS (MAPE $\leq$ 20%). It is 3.99% higher than those of NLIM, and the maximum number of target links accordingly having accurate travel time estimation is approximately 10806 at DST =98% (50% of total DST-Links and 80% of links involved in the experiment (DST =99%)).

Fig. 8 also illustrates a significant drop in the percentage of links having MAPE $\leq$ 20% at DST = 18%. That happens due to the joining of the trunk links and the primary links into the experiment. The temporal and spatial relationships between those links and the motorway links seem to be intricate and not fully captured by the used machine learning techniques.

Focussing closer to the results, the performances of the methods were evaluated for each specific link category. A selected traffic link layout can be modelled by multiple NLIM models. Therefore, it also can be modelled by multiple SMS models. The performances of SMS models and NLIM models on the traffic link layout are compared based on the performance of the best SMS and the best NLIM using the MAPE performance metric.

Fig. 10 presents the relationship between density of the best SMS and the best NLIM models per link type, and their respective MAPE achieved on the unseen data. The SMS uses the temporal and spatial relationship which is modelled by NLIM-RPROP for the searching similar model process.

It can be seen in Fig. 10 that SMS outperforms the NLIM methods in all link types, but especially in A, B and minor roads links. It was observed that the number of minor road links having MAPE less than 12% increased from approximately 50% to above 75%. And the number of B links having

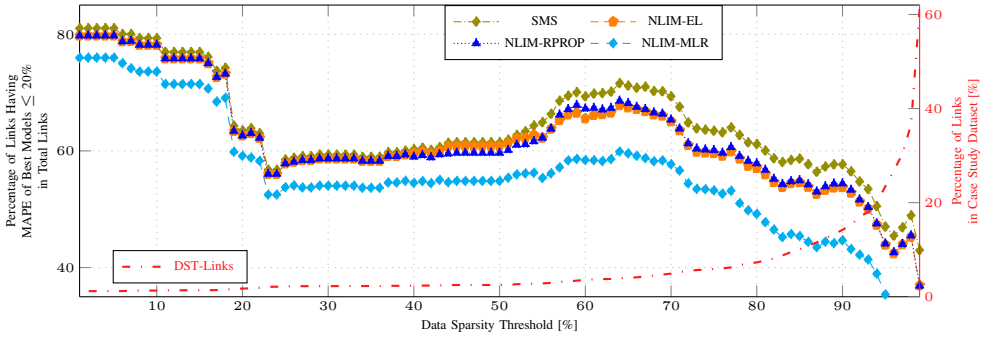


Fig. 8. The percentage of links that have MAPE of the best model less than or equal to 20% against the sparsity threshold achieved by SMS, NLIM-EL, NLIM-RPRO, NLIM-MLR on the unseen data. Note that threshold of MAPE $\leq$ 20% was determined empirically from our preliminary results in [11] and indicates that NLIM or SMS can estimate travel time for more than 75% of the total number of target links in the traffic network.

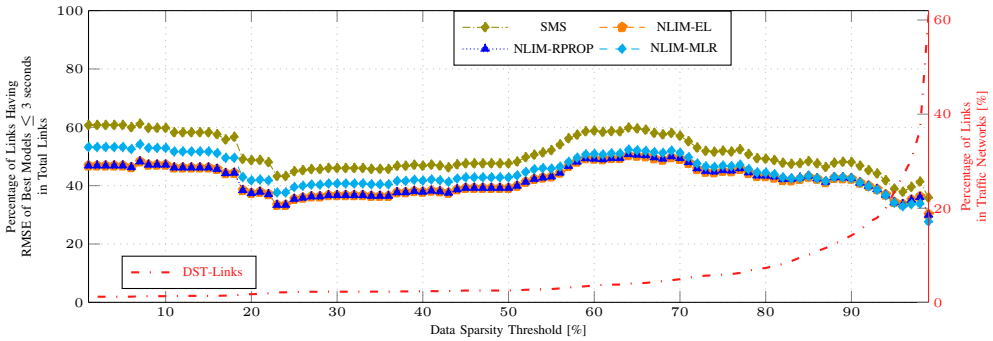


Fig. 9. The percentage of links that have RMSE of the best model less than or equal to 3 seconds against the sparsity threshold achieved by SMS, NLIM-EL, NLIM-RPRO, NLIM-MLR on the unseen data. Note that threshold of RMSE $\leq$ 3s was determined empirically from our preliminary results in [11] and indicates that NLIM or SMS can estimate travel time for more than 75% of the total number of target links in the traffic network.

MAPE less than or equal to 15% also increased from 50% to 75%.

It can be concluded that reinforcement training data from similar NLIM models, support more information for a target NLIM model to learn precisely the spatial and temporal relationship between travel times in traffic links. This technique is especially functional for datasets with variability, irregularity and sparsity which are often characteristics of urban travel time.

V. Conclusion

Improving the performance of travel time models for minor roads which often lack of reliable measurements was considered in this paper. The main idea is to discover traffic link models which are similar to the target traffic link model in order to improve its estimation accuracy. The proposed SMS method has been evaluated on a case study of Leicestershire traffic network in the UK. The NLIM [11] was used to generate a collection of NLIM models subsequently forwarded to the SMS algorithm, which creates the target model using a labelled dataset of similar models together with the target model training dataset.

Results show that SMS method is capable of improving the performance of NLIM on learning the temporal and spatial relationship between the travel time of a target link and travel time of its neighbouring link despite the high sparsity and irregularities in the dataset.

The SMS can increase the amount of training samples for the all link types but the biggest increase was observed in minor links. The number of similar models of each selected traffic link model varies. It ranges from 0 to 10 similar models. The average for the amount of the similar models found by SMS is 2 and 3 for each traffic link category.

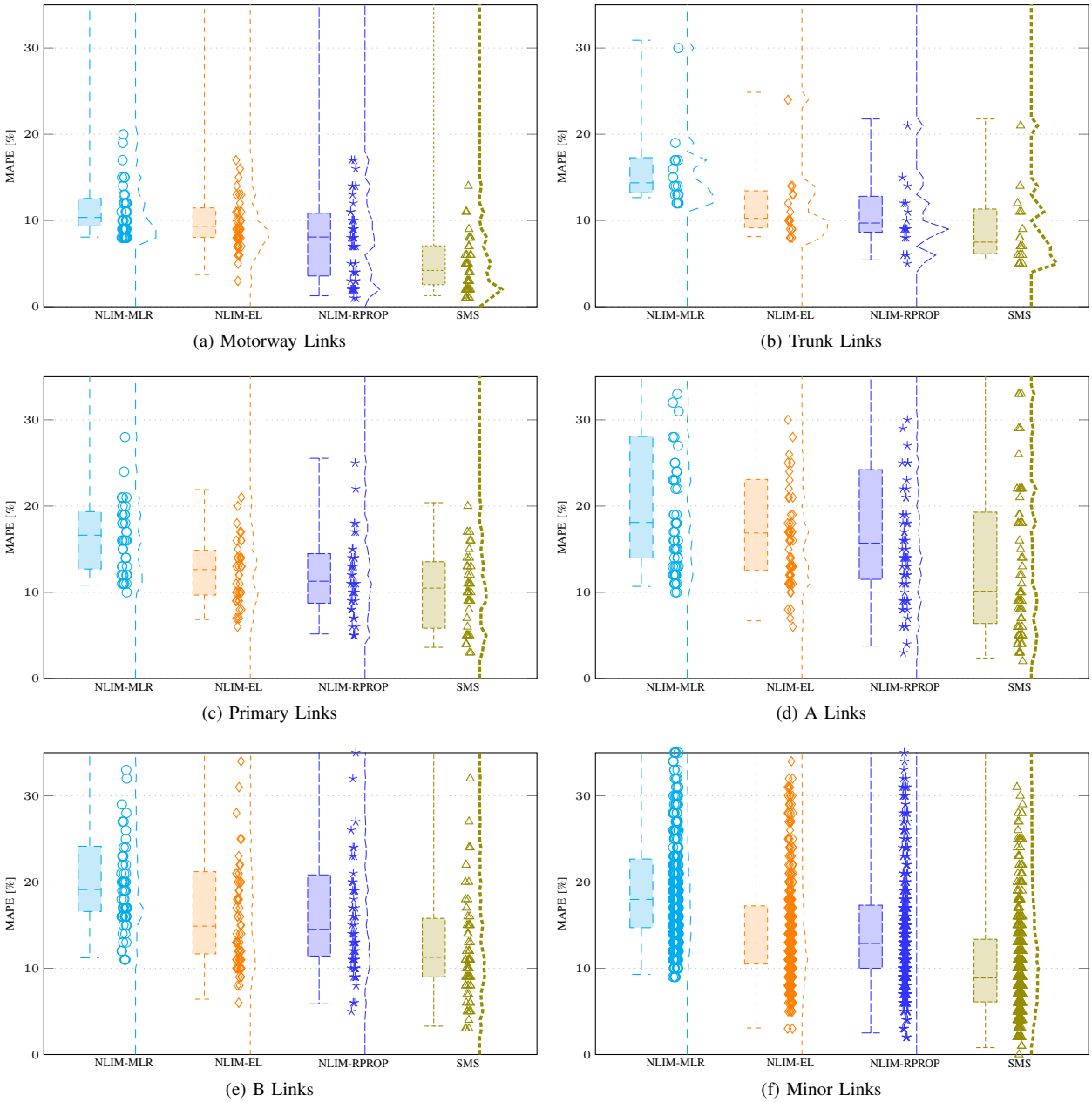
The SMS algorithm outperforms NLIM-MLR, NLIM-EL and NLIM-RPROP on all traffic link categories. The SMS technique works more effectively especially on minor links. 75% of SMS models can produce travel time data which have MAPE error less than 20%. 50% of SMS models can estimate near real-time travel time that has MAPE less than 13.5%, and 25% of SMS models can calculate near real-time travel that has MAPE less than 9.52%.

It can be concluded that reinforcement training data from similar NLIM models provide more information for NLIM to learn the temporal and spatial relationship between the travel time of links supporting the high variability of urban traffic travel time and high data sparsity.

Having more accurate travel time models traffic controllers can make more informative decisions to alleviate traffic congestions. The proposed algorithm can also be embedded into supervisory controllers of autonomous vehicles to improve their route planning capabilities.

Acknowledgment

This work was supported by the Polish National Science Centre under grant no. DEC-2016/21/B/HS4/00667.



(a) Motorway Links (b) Trunk Links

(c) Primary Links (d) A Links

(e) B Links (f) Minor Links

Fig. 10. The density of best models in terms of MAPE for each method per link type presented in a form of a box plot (lower whisker, lower quartile, median, upper quartile, upper whisker). Some high MAPE data points are out of the figure range, hence corresponding upper-whiskers cannot be shown.

REFERENCES

[1] G. Cookson and B. Pishue, "Inrix global traffic scorecard," 2017.

[2] N. Petrovska and A. Stevanovic, "Traffic congestion analysis visualisation tool," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*, 2015, pp. 1489–1494.

[3] D. Capes and R. Hewitt, "Integration improves traffic management in york, uk," *Proceedings of the Institution of Civil Engineers - Municipal Engineer*, vol. 158, no. 4, pp. 275–280, 2005.

[4] K. Tang, S. Chen, and Z. Liu, "Citywide spatial-temporal travel time estimation using big and sparse trajectories," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12, 2018.

[5] G. Kim, "Travel time estimation in vehicle routing problem," in *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2017, pp. 1004–1008.

[6] X. Ma and H. N. Koutsopoulos, "A new online travel time estimation approach using distorted automatic vehicle identification data," in *2008 11th International IEEE Conference on Intelligent Transportation Systems*, 2008, pp. 204–209.

[7] E. Jenelius and H. N. Koutsopoulos, "Travel time estimation for urban road networks using low frequency probe vehicle data," *Transportation Research Part B: Methodological*, vol. 53, pp. 64 – 81, 2013.

[8] M. Jones, Y. Geng, D. Nikovski, and T. Hirata, "Predicting link travel times from floating car data," in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, 2013, pp. 1756–1763.

[9] H. Tu, H. van Lint, and H. V. Zuylen, "Travel time variability versus freeway characteristics," in *2006 IEEE Intelligent Transportation Systems Conference*, 2006, pp. 383–388.

[10] C. P. I. J. V. Hinsbergen, A. Hegyi, J. W. C. V. Lint, and H. J. V. Zuylen, "Bayesian neural networks for the prediction of stochastic travel times in urban networks," *IET Intelligent Transport Systems*, vol. 5, no. 4, pp. 259–265, 2011.

[11] L. H. Vu, B. N. Passow, D. Paluszczyszyn, L. Deka, and E. Goodyer, "Neighbouring link travel time inference method using artificial neural

network," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–8.
[12] U. Department of Transport. (2012, January) Guidance on road classification and the primary route network.
[13] Z. Meng, C. Wang, L. Peng, A. Teng, and T. Z. Qiu, "Link travel time and delay estimation using transit avl data," in *2017 4th International Conference on Transportation Information and Safety (ICTIS)*, 2017, pp. 67–72.
[14] L. Lu, J. Wang, Z. He, and C. Y. Chan, "Real-time estimation of freeway travel time with recurrent congestion based on sparse detector data," *IET Intelligent Transport Systems*, vol. 12, no. 1, pp. 2–11, 2018.
[15] J. M. Ernst, J. V. Krogmeier, and D. M. Bullock, "Estimating required probe vehicle re-identification requirements for characterizing link travel times," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 1, pp. 50–58, 2014.
[16] Q. Guo, C. K. Heng, Y. L. Theng, Y. S. Ong, and P. S. Tan, "Offline time-sensitive travel time estimation in an urban road network," in *2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2015, pp. 848–852.
[17] K. Lee, A. Prokhorchuk, J. Dauwels, and P. Jaillet, "Estimation of travel time from taxi gps data," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–6.
[18] U. Department of Transport, "Congestion (average speed during the weekday morning peak) on local a roads methodology," 2016.
[19] M. Rahmani, E. Jenelius, and H. Koutsopoulos, "Floating car and camera data fusion for non-parametric route travel time estimation," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, 2014, pp. 1286–1291.
[20] K. Vidovi, S. Manduka, and D. Bri, "Estimation of urban mobility using public mobile network," in *2017 International Symposium ELMAR*, 2017, pp. 21–24.
[21] C. D. Chitraranjan, A. M. Denton, and A. S. Perera, "A complete observation model for tracking vehicles from mobile phone signal strengths and its potential in travel-time estimation," in *2016 IEEE 84th Vehicular Technology Conference (VTC-Fall)*, 2016, pp. 1–7.
[22] L. Li, X. Chen, Z. Li, and L. Zhang, "Freeway travel-time estimation based on temporal-spatial queueing model," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 14, no. 3, pp. 1536–1541, 2013.
[23] M. Leodolter, H. Koller, and M. Straub, "Estimating travel times from static map attributes," in *Models and Technologies for Intelligent Transportation Systems (MT-ITS), 2015 International Conference on*, 2015, pp. 121–126.
[24] X. Zhan, S. Hasan, S. V. Ukkusuri, and C. Kamga, "Urban link travel time estimation using large-scale taxi data with partial information," *Transportation Research Part C: Emerging Technologies*, vol. 33, pp. 37 – 49, 2013.
[25] L. Lin, Z. He, and S. Peeta, "Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach," *Transportation Research Part C: Emerging Technologies*, vol. 97, pp. 258–276, 2018.
[26] F. G. Habtemichael and M. Cetin, "Short-term traffic flow rate forecasting based on identifying similar traffic patterns," *Transportation Research Part C: Emerging Technologies*, vol. 66, pp. 61 – 78, 2016, advanced Network Traffic Management: From dynamic state estimation to traffic control.
[27] Z. Zhang, Y. Wang, P. Chen, Z. He, and G. Yu, "Probe data-driven travel time forecasting for urban expressways by matching similar spatiotemporal traffic patterns," *Transportation Research Part C: Emerging Technologies*, vol. 85, pp. 476 – 493, 2017.
[28] L. H. Vu, "Estimation of travel time using temporal and spatial relationships in sparse data," Ph.D. dissertation, 2019. [Online]. Available: http://hdl.handle.net/2086/17512
[29] X. Zhao and J. C. Spall, "Estimating travel time in urban traffic by modeling transportation network systems with binary subsystems," in *2016 American Control Conference (ACC)*, July 2016, pp. 803–808.
[30] J.-P. Rodrigue, C. Comtois, and B. Slack, *The Geography of Transport Systems (3rd Edition)*. Routledge, 2013.
[31] teletracnavman. (2018) teletracnavman. [Online]. Available: https://www.teletracnavman.co.uk/company/press

Luong H. Vu received the B.S. in Information Technology from University of Information and Communication Technology, Thai Nguyen, Viet Nam, in 2007, and the M.Sc. in Information Technology from Batangas State University, Philippines in 2011. In 2019 he was awarded with the Ph.D. from De Montfort University, Leicester, UK. His current interest include intelligent transport systems and computational intelligence.

Benjamin N. Passow is a Research and Development Engineer, ITK Engineering GmbH, Germany, and a part-time lecturer in Computational Intelligence and Robotics at De Montfort University in Leicester, UK. He received the M.Sc. degree in 2007 and the Ph.D. degree in 2011, both in Computational Intelligence and Robotics from De Montfort University. He is research active within the fields of AI/CI, Transport and Robotics and has participated in various UK, EU and ESA funded projects. In 2009 he received the Machine Intelligence Award from the British Computer Society in Cambridge. His research interests include the theory and application of computational intelligence in intelligent transport, acoustic sensing, autonomous mobile robots, unmanned aerial vehicles, and embedded systems.

Daniel Paluszczyszyn is a Senior Lecturer in the School of Engineering and Sustainable Development, De Montfort University, UK. His recent research interests consider various aspects of intelligent mobility including optimisation of the energy management system for low carbon vehicles and scheduling approaches to charge autonomous electric vehicles. He received the B.Eng. in Computer Engineering from the University of Zielona Gora, Poland, in 2003, and the M.Sc. in Systems and Control from the Coventry University, UK, in 2008. In 2015 he was awarded with the Ph.D. in Hydroinformatics from the De Montfort University.

Lipika Deka is a Computer Engineer, currently conducting research within the field of Intelligent Transport Systems and Connected Autonomous Cars. With an educational background in Computer Science and Engineering, PhD in the field of Concurrency Control within File Systems, she has interdisciplinary research experience of applying Computer Science algorithms (mainly AI techniques) in the fields of Chemical Engineering, Healthcare Infrastructure and Intelligent Transport Systems. Presently, she is working on the Software Engineering, Software update, positioning accuracy and path planning aspects of Connected Autonomous Cars.

Eric Goodyer is a Professor of Instrumentation. His specialist area of expertise is the design and development of real time instrumentation for industrial, automotive and laboratory use. This is best summed up as Measurement Automation and Control. His research interests are the development of new electromechanical and electronic measuring equipment, and telematic solutions for industry. He works part time for De Montfort University. He also works as a self employed freelance design engineer in my specialist fields, offering design and consultancy service to industry world-wide.